

WHAT IS CLAIMED IS:

1. ~~A method of operating a computing device, comprising:~~

accessing a JavaScript program, said JavaScript program including a sequence of JavaScript instructions;

parsing said JavaScript program to validate said JavaScript instructions;

generating an intermediate representation of said JavaScript instructions of said JavaScript program;

interfacing with said JavaScript program's library of host objects; and

executing said intermediate representation to produce the results intended by said JavaScript program;

wherein said generating is implemented in the Java programming language, wherein said intermediate representation represents the JavaScript program instructions, and wherein said intermediate representation is suitable for execution by a JavaScript interpreter implemented in the Java programming language.

2. The method of claim 1, wherein said intermediate representation is further suitable for execution by a stack-machine implemented interpreter.

3. The method of claim 1, wherein said intermediate representation scheme ~~uses Java classes and objects, and creates logical commands representing said JavaScript program instructions.~~

09714051-11500

00577-1504260

4. ~~The method of claim 1, wherein said generating comprises using a stack~~
data structure to store said intermediate representation as Java objects.

5. The method of claim 4, wherein said stack data structure is constructed from an array of elements, wherein each said element is either an operator or an operand, and wherein each said element derives from an abstract Java class with generic functionality and each said element contains data relating to the operator or data type it represents.

6. An information carrier medium that, when engaged in operable relation to a computer system, is configured to communicate to the system a representation of a software program, wherein the software program comprises instructions to interpret and execute programs by:

accessing a JavaScript program, said JavaScript program including a sequence of JavaScript instructions;

parsing said JavaScript program to validate said instructions;

generating an intermediate representation of said JavaScript instructions of said JavaScript program;

interfacing with said JavaScript program's library of host objects; and

executing said intermediate representation to produce the results intended by said JavaScript program;

wherein said generating is implemented in the Java programming language,
~~wherein said intermediate representation represents the JavaScript~~

~~program instructions, and wherein said intermediate representation is~~
suitable for execution by a JavaScript interpreter.

7. The medium of claim 6, wherein said intermediate representation is suitable for execution by a stack-machine implemented interpreter.

8. The medium of claim 6, wherein said intermediate representation scheme uses Java classes and objects, and creates logical commands representing said JavaScript program instructions.

9. The medium of claim 6, wherein said generating comprises using a stack data structure to store said intermediate representation as Java objects.

10. The medium of claim 9, wherein said stack data structure is constructed from an array of elements, wherein each said element is either an operator or an operand, and wherein each said element derives from an abstract Java class with generic functionality and each said element contains data relating to the operator or data type it represents.

Add
AI